

РАЗРАБОТКА ПРИЛОЖЕНИЙ НА ПЛАТФОРМЕ ANGULARJS

Н.В. Харитонов, А.С. Дроздов, 2 курс

Научный руководитель – М.В. Стержанов, к.т.н., доцент

Белорусский государственный университет информатики и радиоэлектроники

В современном мире, где разработка программного обеспечения развивается быстрыми темпами, наибольшую популярность приобретают веб–приложения. Основными плюсами использования веб–приложений являются простота, удобство, доступность и скорость. Для реализации таких приложений существует множество различных фреймворков разработки, среди которых одним из наиболее популярных в последние три года является AngularJS.

AngularJS – это проект с открытым исходным кодом, который можно найти на GitHub, и лицензируется компанией Google, Inc. на условиях лицензии MIT. 15 сентября 2016 года состоялся релиз второй версии данного фреймворка, содержащий в себе кардинальные отличия.

Angular 2 – фреймворк от компании Google, который позволяет разрабатывать сложные SPA-приложения (Single Page Application). Написание приложения состоит из создания шаблонов с помощью языка разметки HTML, написания классов-компонентов для работы с данными шаблонами, а также выделения общей логики приложения в сервисы. Все вместе это можно объединять в отдельные модули. Angular 2 написан в соответствии с шаблоном проектирования «модуль», что позволяет изолировать части логики приложения от глобального контекста, и содержит собственную систему модулей (NgModules) с тем, чтобы приложение имело понятную структуру и позволяло переиспользовать необходимые части без циклических зависимостей. Angular 2 написан на языке программирования TypeScript, разработанным компанией Microsoft в 2012 г., и являющимся надмножеством над языком JavaScript. Angular 2 предоставляет возможность написания веб-приложения с помощью следующих языков программирования [1, с. 456][2]:

- TypeScript;
- Dart;
- JavaScript (ES5/ES6).

Данный фреймворк позволяет реализовывать не только веб-приложения, но также мобильные и десктопные приложения, что делает его кроссплатформенным.

Таким образом, основными компонентами Angular2, увеличивающими уровень абстракции, являются:

- модули(Modules);
- компоненты(Components)–управляют отображением данных в приложении;
- шаблоны (Templates) – используются для определения представлений приложения;
- метаданные (Metadata) – описания поведения компонентов и классов;
- привязка данных(Data binding)–удобный способ организации взаимодействия шаблонов и компонентов, представлен в трех видах one–time binding (предоставление данных из модели в представление, без отслеживания изменений «на лету»), one–way binding (позволяющий отслеживать изменения в модели и отображать их в представлении), two– way binding (позволяющий отслеживать изменения как в модели и отображать их в представлении, так и наоборот);
- директивы (Directives) – компоненты для расширения возможностей шаблонизатора;
- сервисы (Services) – используются для создания компонентов, которые могут быть доступны для всего приложения
- внедрение зависимостей (Dependency injection) – обеспечение экземпляры компонентов внешними зависимостями;
- роутер (Router) – компонент для обеспечения навигации по приложению;
- формы (Forms) – компонент для работы с вводимыми пользователем данными;
- анимация (Animations) – компоненты для анимации пользовательского интерфейса.

Традиционно, веб-приложение на Angular 2 состоит из набора компонентов (виджетов), которые образуют древовидную структуру. В ней, в свою очередь, родительский компонент имеет ссылку на все вложенные в него дочерние. Обмен данными реализован следующим способом. При получении данных родитель отправляет их часть дочерним компонентам. Они, в свою очередь, либо передают их часть дальше по дереву компонентов, либо отображают эти данные с использованием различных элементов интерфейса. Также, в Angular 2 дочерние компоненты имеют возможность оповестить родительский о различных пользовательских событиях: клик мыши или нажатие клавиши. Также возможно внедрение так называемых сервисов (services) – специальных объектов, предоставляющих компоненту возможность получить данные в любой момент, вне зависимости от его расположения в дереве компонентов. Стоит упомянуть, что в Angular 2 реализован прием, позволяющий любому объекту веб-приложения получить доступ к другому, предварительно зарегистрированному объекту (provider) при необходимости.

Приложения, написанные с помощью Angular 2, легко тестировать не только с помощью специальной утилиты Angular Testing Utilities, создающей тестовое окружение для приложения, но и с помощью широко известных фреймворков для тестирования, таких как Jasmine, Karma и Protractor.

Таким образом, фреймворк Angular.js является полноценным инструментом, полноценно реализующим компонентный подход, позволяя разработчикам создавать отдельные компоненты, а из компонентов создавать полноценные приложения. Созданные компоненты можно использоваться в других приложениях, так как они независимы.

Список использованных источников

1. Fain Y., Moiseev A. Angular 2 Development with TypeScript. – Manning Publications, 2016
2. Angular 2 [Электронный ресурс]. – Режим доступа: <https://angular.io/>